


```

LL          IIIII
LL          IIIII
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LLLLLLLLLLLL IIIII
LLLLLLLLLLLL IIIII

          SSSSSSSS
          SSSSSSSS
          SS
          SS
          SS
          SS
          SSSSSS
          SSSSSS
          SS
          SS
          SS
          SS
          SSSSSSSS
          SSSSSSSS
          SSSSSSSS
          SSSSSSSS

```



```
0001 0 %TITLE 'EDT$FIOHNDLR - file I/O condition handler'
0002 0 MODULE EDT$FIOHNDLR ( ! file I/O condition hanndler
0003 0 IDENT = 'V04-000' ! File: FIOHNDLR.B32 Edit: JBS1016
0004 0 ) =
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0011 1 * ALL RIGHTS RESERVED.
0012 1 *
0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0018 1 * TRANSFERRED.
0019 1 *
0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0022 1 * CORPORATION.
0023 1 *
0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0026 1 *
0027 1 *
0028 1 *****
0029 1
0030 1
0031 1 ++
0032 1 FACILITY: EDT -- The DEC Standard Editor
0033 1
0034 1 ABSTRACT:
0035 1
0036 1 Condition handler enabled for all file I/O
0037 1
0038 1 ENVIRONMENT: Runs on Vax/Vms
0039 1
0040 1 AUTHOR: Shelly T. Solomon, CREATION DATE: 04-Jan-1982
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. STS 04-Jan-1982
0045 1 1-002 - Fixed command file error. STS 14-Jan-1982
0046 1 1-003 - Catch open errors. STS 25-Feb-1982
0047 1 1-004 - Add literals for callablre parameters. STS 08-Mar-1982
0048 1 1-005 - Change EDT$$G_CMD to EDT$$V_OPTIONS. JBS 18-Mar-1982
0049 1 1-006 - If we unwind, make sure there is a bad status code in R0. JBS 26-mar-1982
0050 1 1-007 - Revise offsets into the signal args to reflect the new signaling
0051 1 in FILEIO -- the RMS STS is now always in the fourth longword. JBS 29-Mar-1982
0052 1 1-008 - Put in lots more error checking. JBS 29-Mar-1982
0053 1 1-009 - Catch read and write errors and unwind. STS 14-Jul-1982
0054 1 1-010 - Set abort write flag if error on get. STS 05-Aug-1982
0055 1 1-011 - Make use of bit masks for options word. STS 17-Aug-1982
0056 1 1-012 - Fix up references to EDT$M symbols. JBS 23-Aug-1982
0057 1 1-013 - Don't alter returned R0 when unwinding. JBS 23-Feb-1983
```


EDT\$FIOHNDLR
V04-000

EDT\$FIOHNDLR - file I/O condition handler

C 1
16-Sep-1984 00:25:45
14-Sep-1984 12:23:08

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]FIOHNDLR.B32;1 Page 2 (1)

```
: 58      0058 1 : 1-014 - If we resignal, arrange to rebuild the screen if we are continued. JBS 22-Mar-1983
: 59      0059 1 : 1-015 - Remove some unreachable code. JBS 23-Mar-1983
: 60      0060 1 : 1-016 - Improve the appearance of the listing. JBS 14-Jun-1983
: 61      0061 1 : --
: 62      0062 1 :
```

EDT
V04

: R

:
:
:

EDTSFIOHNDLR
V04-000

EDTSFIOHNDLR - file I/O condition handler
Declarations

D 1
16-Sep-1984 00:25:45
14-Sep-1984 12:23:08

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]FIOHNDLR.B32;1
Page 3 (2)

```
: 64      0063 1 %SBTTL 'Declarations'
: 65      0064 1
: 66      0065 1 TABLE OF CONTENTS:
: 67      0066 1
: 68      0067 1
: 69      0068 1 REQUIRE 'EDTSRC:TRAROUNAM';
: 70      0507 1
: 71      0508 1 FORWARD ROUTINE
: 72      0509 1     EDT$FIOHNDLR;
: 73      0510 1
: 74      0511 1
: 75      0512 1 INCLUDE FILES:
: 76      0513 1
: 77      0514 1
: 78      0515 1 REQUIRE 'EDTSRC:EDTREQ';
: 79      0650 1
: 80      0651 1 REQUIRE 'EDTSRC:SYSSYM';
: 81      0681 1
: 82      0682 1
: 83      0683 1 MACROS:
: 84      0684 1
: 85      0685 1     NONE
: 86      0686 1
: 87      0687 1 EQUATED SYMBOLS:
: 88      0688 1
: 89      0689 1     NONE
: 90      0690 1
: 91      0691 1 FIELDS:
: 92      0692 1
: 93      0693 1     NONE
: 94      0694 1
: 95      0695 1 OWN STORAGE:
: 96      0696 1
: 97      0697 1     in the routine
: 98      0698 1
: 99      0699 1 EXTERNAL REFERENCES:
: 100     0700 1
: 101     0701 1     in the routine
: 102     0702 1
```

```
104 0703 1 %SBTTL 'EDT$$FIOHNDLR - condition handler for file I/O'
105 0704 1
106 0705 1 GLOBAL ROUTINE EDT$$FIOHNDLR (
107 0706 1     SIG,
108 0707 1     MECH,
109 0708 1     ENBL
110 0709 1 ) =
111 0710 1
112 0711 1 ++
113 0712 1 FUNCTIONAL DESCRIPTION:
114 0713 1
115 0714 1     This is the condition handler for file I/O. It intercepts
116 0715 1     the non-fatal conditions, such as file not found, and unwinds
117 0716 1     to the caller of EDT$$CALLFIO. Errors not recognized are
118 0717 1     resigaled to the caller to EDT$EDIT. Just in case that caller
119 0718 1     has a handler which does a continue, arrange to reset and rebuild
120 0719 1     the screen if the screen updater regains control, and stop the
121 0720 1     working message so it won't interfere with whatever the user
122 0721 1     may have in mind.
123 0722 1
124 0723 1 FORMAL PARAMETERS:
125 0724 1
126 0725 1     Usual for condition handlers.
127 0726 1
128 0727 1 IMPLICIT INPUTS:
129 0728 1
130 0729 1     NONE
131 0730 1
132 0731 1 IMPLICIT OUTPUTS:
133 0732 1
134 0733 1     EDT$$G_IOFI_NFND      1 = error was "file not found"
135 0734 1     EDT$$G_ABT_WRITE      1 = abort the current WRITE or EXIT operation.
136 0735 1     EDT$$G_SCR_CHGD      2 = reset and rebuild screen on continue
137 0736 1
138 0737 1 COMPLETION STATUS:
139 0738 1
140 0739 1     As usual for condition handlers
141 0740 1
142 0741 1 SIDE EFFECTS:
143 0742 1
144 0743 1     NONE
145 0744 1
146 0745 1 --
147 0746 1
148 0747 2 BEGIN
149 0748 2
150 0749 2 EXTERNAL ROUTINE
151 0750 2     EDT$$STOP_WKINGMSG : NOVALUE;
152 0751 2
153 0752 2 EXTERNAL
154 0753 2     EDT$$G_ABT_WRITE,
155 0754 2     EDT$$G_IO_FNAM,
156 0755 2     EDT$$G_IO_STS,
157 0756 2     EDT$$G_IO_STV,
158 0757 2     EDT$$G_IOFI_NFND,
159 0758 2     EDT$$G_SCR_REBUILD;
160 0759 2
```



```
161 0760 2    EXTERNAL LITERAL
162 0761 2    EDT$K_COMMAND_FILE,
163 0762 2    EDT$K_INPUT_FILE,
164 0763 2    EDT$K_INCLUDE_FILE,
165 0764 2    EDT$K_WRITE_FILE,
166 0765 2    EDT$K_OUTPUT_FILE,
167 0766 2    EDT$K_PUT,
168 0767 2    EDT$K_GET,
169 0768 2    EDT$K_OPEN_INPUT,
170 0769 2    EDT$K_OPEN_OUTPUT_NOSEQ,
171 0770 2    EDT$K_OPEN_OUTPUT_SEQ;
172 0771 2
173 0772 2    MAP
174 0773 2    SIG : REF VECTOR,
175 0774 2    MECH : REF VECTOR,
176 0775 2    ENBL : REF VECTOR;
177 0776 2
178 0777 2    BIND
179 0778 2    FCODE = .ENBL [2],
180 0779 2    RET_VALUE = MECH [3],
181 0780 2    STREAM = .ENBL [1];
182 0781 2
183 0782 2    LOCAL
184 0783 2    STATUS;
185 0784 2
186 0785 2    IF ((.SIG [0] EQL 1) AND (.SIG [1] EQL SS$_UNWIND))
187 0786 2    THEN
188 0787 2    BEGIN
189 0788 2    +
190 0789 2    - We are unwinding.
191 0790 2    -
192 0791 2    RETURN (SS$_CONTINUE);
193 0792 2    END;
194 0793 2
195 0794 2    IF ((.SIG [0] NEQ 7) OR (.SIG [2] NEQ 1))
196 0795 2    THEN
197 0796 2    BEGIN
198 0797 2    +
199 0798 2    - If we are continued, reset and rebuild the screen. There is no telling
200 0799 2    - what a user might do in his own handler before continuing us.
201 0800 2    -
202 0801 2    EDT$$G_SCR_REBUILD = 2;
203 0802 2    +
204 0803 2    - Also, stop the working message, so it won't interfere with whatever the
205 0804 2    - user may have in mind.
206 0805 2    -
207 0806 2    EDT$$STOP_WKINGMSG ();
208 0807 2    RETURN (SS$_RESIGNAL);
209 0808 2    END;
210 0809 2
211 0810 2    +
212 0811 2    - This is probably a signal from the file I/O routine.
213 0812 2    - Set up the message vector for the call to EDT$$IO_ERRMSG
214 0813 2    - and save the filename signalled with
215 0814 2    -
216 0815 2    EDT$$G_IO_STS = .SIG [4];
217 0816 2    EDT$$G_IO_STV = .SIG [5];
```

```
218 0817 2 EDT$$A_IO_FNAM = .SIG [3];
219 0818 2
220 0819 2 SELECTONE .STREAM OF
221 0820 2 SET
222 0821 2
223 0822 2 [EDT$K_COMMAND_FILE] :
224 0823 3 BEGIN
225 0824 3
226 0825 4 IF (.SIG [4] EQL RMS$_FNF) ! File not found
227 0826 3 THEN
228 0827 4 BEGIN
229 0828 4 EDT$$G_IOFI_NFND = 1; ! set file not found flag
230 0829 4 SETUNWIND (); ! everything O.K., continue (caller will print message)
231 0830 4 RETURN (SS$_CONTINUE);
232 0831 4 END
233 0832 3 ELSE
234 0833 3 !+
235 0834 3 Not "file not found", resignal the error.
236 0835 3 -
237 0836 4 BEGIN
238 0837 4 !+
239 0838 4 If we are continued, reset and rebuild the screen. There is no telling
240 0839 4 what a user might do in his own handler before continuing us.
241 0840 4 -
242 0841 4 EDT$$G_SCR_REBUILD = 2;
243 0842 4 !+
244 0843 4 Also, stop the working message, so it won't interfere with whatever the
245 0844 4 user may have in mind.
246 0845 4 -
247 0846 4 EDT$$STOP_WKINGMSG ();
248 0847 4 RETURN (SS$_RESIGNAL);
249 0848 3 END;
250 0849 3
251 0850 2 END;
252 0851 2
253 0852 2 [EDT$K_INPUT_FILE] :
254 0853 3 BEGIN
255 0854 3
256 0855 4 IF (.SIG [4] EQL RMS$_FNF)
257 0856 3 THEN
258 0857 4 BEGIN
259 0858 4 EDT$$G_IOFI_NFND = 1; ! set file not found flag
260 0859 4 SETUNWIND (); ! everything O.K., continue
261 0860 4 RETURN (SS$_CONTINUE);
262 0861 4 END
263 0862 3 ELSE
264 0863 3
265 0864 4 IF (.FCODE EQL EDT$K_GET)
266 0865 3 THEN
267 0866 4 BEGIN
268 0867 4 EDT$$G_ABT_WRITE = 1;
269 0868 4 SETUNWIND ();
270 0869 4 RETURN (SS$_CONTINUE);
271 0870 4 END
272 0871 3 ELSE
273 0872 4 BEGIN
274 0873 4 !+
```



```

275 0874 4 | If we are continued, reset and rebuild the screen. There is no telling
276 0875 4 | what a user might do in his own handler before continuing us.
277 0876 4 | -
278 0877 4 |     EDT$$G_SCR_REBUILD = 2;
279 0878 4 | +
280 0879 4 | Also, stop the working message, so it won't interfere with whatever the
281 0880 4 | user may have in mind.
282 0881 4 | -
283 0882 4 |     EDT$$STOP_WKINGMSG ();
284 0883 4 |     RETURN (SS$_RESIGNAL);
285 0884 4 |     END;
286 0885 3 |
287 0886 2 |     END;
288 0887 2 |
289 0888 2 | [EDT$K_INCLUDE_FILE, EDT$K_WRITE_FILE, EDT$K_OUTPUT_FILE] :
290 0889 3 | BEGIN
291 0890 3 | +
292 0891 3 | Any error opening any of the above files is not considered fatal.
293 0892 3 | -
294 0893 3 |
295 0894 4 |     IF ((.FCODE EQL EDT$K_OPEN_INPUT) OR
296 0895 4 |         (.FCODE EQL EDT$K_OPEN_OUTPUT_NOSEQ) OR
297 0896 4 |         (.FCODE EQL EDT$K_OPEN_OUTPUT_SEQ))
298 0897 3 |     THEN
299 0898 4 |         BEGIN
300 0899 4 | +
301 0900 4 | If the error is "file not found" (only possible with INCLUDE) then
302 0901 4 | give a more-specific message.
303 0902 4 | -
304 0903 4 |
305 0904 4 |     IF (.SIG [4] EQL RMS$_FNF) THEN EDT$$G_IOFI_NFND = 1;
306 0905 4 |
307 0906 4 |     SETUNWIND ();
308 0907 4 |     RETURN (SS$_CONTINUE);
309 0908 4 |     END
310 0909 3 | ELSE
311 0910 3 |
312 0911 4 |     IF ((.FCODE EQL EDT$K_PUT) OR (.FCODE EQL EDT$K_GET))
313 0912 3 |     THEN
314 0913 4 |         BEGIN
315 0914 4 |             EDT$$G_ABT_WRITE = 1;
316 0915 4 |             SETUNWIND ();
317 0916 4 |             RETURN (SS$_CONTINUE);
318 0917 4 |             END
319 0918 3 |         ELSE
320 0919 4 |             BEGIN
321 0920 4 | +
322 0921 4 | If we are continued, reset and rebuild the screen. There is no telling
323 0922 4 | what a user might do in his own handler before continuing us.
324 0923 4 | -
325 0924 4 |     EDT$$G_SCR_REBUILD = 2;
326 0925 4 | +
327 0926 4 | Also, stop the working message, so it won't interfere with whatever the
328 0927 4 | user may have in mind.
329 0928 4 | -
330 0929 4 |     EDT$$STOP_WKINGMSG ();
331 0930 4 |     RETURN (SS$_RESIGNAL);
```

```
0931      END;
0932
0933      END;
0934
0935      [OTHERWISE] :
0936      BEGIN
0937      +
0938      If we are continued, reset and rebuild the screen. There is no telling
0939      what a user might do in his own handler before continuing us.
0940      -
0941      EDT$$G_SCR_REBUILD = 2;
0942      +
0943      Also, stop the working message, so it won't interfere with whatever the
0944      user may have in mind.
0945      -
0946      EDT$$STOP_WKINGMSG ();
0947      RETURN (SS$_RESIGNAL);
0948      END;
0949      TES;
0950
0951      END;
```

! of routine EDT\$\$FIOHNDLR

.TITLE EDT\$FIOHNDLR EDT\$FIOHNDLR - file I/O condition
handler

.IDENT \V04-000\

```
.EXTRN EDT$$STOP_WKINGMSG
.EXTRN EDT$$G_ABX_WRITE
.EXTRN EDT$$A_IO_FNAM, EDT$$G_IO_STS
.EXTRN EDT$$G_IO_STV, EDT$$G_IOFI_NFND
.EXTRN EDT$$G_SCR_REBUILD
.EXTRN EDT$K_COMMAND_FILE
.EXTRN EDT$K_INPUT_FILE
.EXTRN EDT$K_INCLUDE_FILE
.EXTRN EDT$K_WRITE_FILE
.EXTRN EDT$K_OUTPUT_FILE
.EXTRN EDT$K_PUT, EDT$K_GET
.EXTRN EDT$K_OPEN_INPUT
.EXTRN EDT$K_OPEN_OUTPUT_NOSEQ
.EXTRN EDT$K_OPEN_OUTPUT_SEQ
```

.PSECT _EDT\$CODE, NOWRT, SHR, PIC, 2

.ENTRY EDT\$\$FIOHNDLR, Save R2, R3, R4

```
MOVL #EDT$K_GET, R4
MOVL ENBL, R3
MOVL SIG, R2
CMPL (R2), #1
BNEQ 1$
CMPL 4(R2), #2336
BNEQ 1$
BRW 13$
CMPL (R2), #7
BNEQ 6$
CMPL 8(R2), #1
BNEQ 6$
```

```
001C 00000
54 00000000G 8F D0 00002
53 0C AC D0 00009
52 04 AC D0 0000D
01 62 D1 00011
0D 12 00014
00000920 8F 04 A2 D1 00016
03 12 0001E
00C2 31 00020
07 62 D1 00023 1$:
72 12 00026
01 08 A2 D1 00028
6C 12 0002C
```

: 0705

: 0778

: 0785

: 0794

00000000G	00	10	A2	D0	0002E	MOVL	16(R2), EDT\$\$G_IO_STS	0815
00000000G	00	14	A2	D0	00036	MOVL	20(R2), EDT\$\$G_IO_STV	0816
00000000G	00	0C	A2	D0	0003E	MOVL	12(R2), EDT\$\$A_IO_FNAM	0817
	50	04	B3	D0	00046	MOVL	24(R3), R0	0819
00000000G	8F		50	D1	0004A	CMPL	R0, #EDT\$K_COMMAND_FILE	0822
			0C	12	00051	BNEQ	2\$	
00018292	8F	10	A2	D1	00053	CMPL	16(R2), #98962	0825
			76	12	0005B	BNEQ	10\$	
			13	11	0005D	BRB	3\$	0828
00000000G	8F		50	D1	0005F	CMPL	R0, #EDT\$K_INPUT_FILE	0852
			19	12	00066	BNEQ	5\$	
00018292	8F	10	A2	D1	00068	CMPL	16(R2), #98962	0855
			09	12	00070	BNEQ	4\$	
00000000G	00		01	D0	00072	MOVL	#1, EDT\$\$G_IOFI_NFND	0858
			61	11	00079	BRB	12\$	0859
	54	08	B3	D1	0007B	CMPL	28(R3), R4	0864
			52	11	0007F	BRB	10\$	
00000000G	8F		50	D1	00081	CMPL	R0, #EDT\$K_INCLUDE_FILE	0888
			12	13	00088	BEQL	7\$	
00000000G	8F		50	D1	0008A	CMPL	R0, #EDT\$K_WRITE_FILE	
			09	13	00091	BEQL	7\$	
00000000G	8F		50	D1	00093	CMPL	R0, #EDT\$K_OUTPUT_FILE	
			4D	12	0009A	BNEQ	14\$	
	50	08	B3	D0	0009C	MOVL	28(R3), R0	0894
00000000G	8F		50	D1	000A0	CMPL	R0, #EDT\$K_OPEN_INPUT	
			12	13	000A7	BEQL	8\$	
00000000G	8F		50	D1	000A9	CMPL	R0, #EDT\$K_OPEN_OUTPUT_NOSEQ	0895
			09	13	000B0	BEQL	8\$	
00000000G	8F		50	D1	000B2	CMPL	R0, #EDT\$K_OPEN_OUTPUT_SEQ	0896
			0C	12	000B9	BNEQ	9\$	
00018292	8F	10	A2	D1	000BB	CMPL	16(R2), #98962	0904
			17	12	000C3	BNEQ	12\$	
			AB	11	000C5	BRB	3\$	
00000000G	8F		50	D1	000C7	CMPL	R0, #EDT\$K_PUT	0911
			05	13	000CE	BEQL	11\$	
	54		50	D1	000D0	CMPL	R0, R4	
			14	12	000D3	BNEQ	14\$	
00000000G	00		01	D0	000D5	MOVL	#1, EDT\$\$G_ABT_WRITE	0914
			7E	7C	000DC	CLRQ	-(SP)	0915
00000000G	00		02	FB	000DE	CALLS	#2, SYSSUNWIND	
	50		01	D0	000E5	MOVL	#1, R0	0916
				04	000E8	RET		
00000000G	00		02	D0	000E9	MOVL	#2, EDT\$\$G_SCR_REBUILD	0941
00000000G	00		00	FB	000F0	CALLS	#0, EDT\$\$STOP_WKINGMSG	0946
	50	0918	8F	3C	000F7	MOVZWL	#2328, R0	0947
				04	000FC	RET		0951

; Routine Size: 253 bytes, Routine Base: _EDT\$CODE + 0000

; 353 0952 1
; 354 0953 1 !<BLF/PAGE>

EDT\$FIOHNDLR
V04-000

EDT\$FIOHNDLR - file I/O condition handler
EDT\$FIOHNDLR - condition handler for file I/O

K 1
16-Sep-1984 00:25:45
14-Sep-1984 12:23:08

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]FIOHNDLR.B32;1

Page 10
(4)

: 356 0954 1 END
: 357 0955 1
: 358 0956 0 ELUDOM

! of module EDT\$FIOHNDLR

.EXTRN SYSSUNWIND

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	253	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
-\$255\$DUA28:[EDT.SRC]EDT.L32;1	377	0	0	40	00:00.2
-\$255\$DUA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1
-\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	4	0	581	00:04.0

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS\$:FIOHNDLR/OBJ=OBJ\$:FIOHNDLR MSRC\$:FIOHNDLR.B32/UPDATE=(ENH\$:FIOHNDLR)

: Size: 253 code + 0 data bytes
: Run Time: 00:23.1
: Elapsed Time: 00:30.9
: Lines/CPU Min: 2486
: Lexemes/CPU-Min: 7427
: Memory Used: 123 pages
: Compilation Complete

0133 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

EXTEND
LIS

FDEC
LIS

FILL
LIS

FINDPARA
LIS

FCRLF
LIS

EDT
LIS

EXEC
LIS

EXECNOO
LIS

FILEIO
LIS

EDTVECTOR
LIS

FINDKEY
LIS

FCOLINC
LIS

FINAL
LIS

FINDHDLR
LIS

DEFKEY
LIS

ERRMSG
LIS

FCHAR
LIS

0134 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

0134	0135	0136	0137	0138	0139	0140	0141	0142	0143	0144	0145	0146	0147	0148	0149	0150	0151	0152	0153	0154	0155	0156	0157	0158	0159	0160	0161	0162	0163	0164	0165	0166	0167	0168	0169	0170	0171	0172	0173	0174	0175	0176	0177	0178	0179	0180	0181	0182	0183	0184	0185	0186	0187	0188	0189	0190	0191	0192	0193	0194	0195	0196	0197	0198	0199																																		
0200	0201	0202	0203	0204	0205	0206	0207	0208	0209	0210	0211	0212	0213	0214	0215	0216	0217	0218	0219	0220	0221	0222	0223	0224	0225	0226	0227	0228	0229	0230	0231	0232	0233	0234	0235	0236	0237	0238	0239	0240	0241	0242	0243	0244	0245	0246	0247	0248	0249	0250	0251	0252	0253	0254	0255	0256	0257	0258	0259	0260	0261	0262	0263	0264	0265	0266	0267	0268	0269	0270	0271	0272	0273	0274	0275	0276	0277	0278	0279	0280	0281	0282	0283	0284	0285	0286	0287	0288	0289	0290	0291	0292	0293	0294	0295	0296	0297	0298	0299
0300	0301	0302	0303	0304	0305	0306	0307	0308	0309	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	0330	0331	0332	0333	0334	0335	0336	0337	0338	0339	0340	0341	0342	0343	0344	0345	0346	0347	0348	0349	0350	0351	0352	0353	0354	0355	0356	0357	0358	0359	0360	0361	0362	0363	0364	0365	0366	0367	0368	0369	0370	0371	0372	0373	0374	0375	0376	0377	0378	0379	0380	0381	0382	0383	0384	0385	0386	0387	0388	0389	0390	0391	0392	0393	0394	0395	0396	0397	0398	0399
0400	0401	0402	0403	0404	0405	0406	0407	0408	0409	0410	0411	0412	0413	0414	0415	0416	0417	0418	0419	0420	0421	0422	0423	0424	0425	0426	0427	0428	0429	0430	0431	0432	0433	0434	0435	0436	0437	0438	0439	0440	0441	0442	0443	0444	0445	0446	0447	0448	0449	0450	0451	0452	0453	0454	0455	0456	0457	0458	0459	0460	0461	0462	0463	0464	0465	0466	0467	0468	0469	0470	0471	0472	0473	0474	0475	0476	0477	0478	0479	0480	0481	0482	0483	0484	0485	0486	0487	0488	0489	0490	0491	0492	0493	0494	0495	0496	0497	0498	0499
0500	0501	0502	0503	0504	0505	0506	0507	0508	0509	0510	0511	0512	0513	0514	0515	0516	0517	0518	0519	0520	0521	0522	0523	0524	0525	0526	0527	0528	0529	0530	0531	0532	0533	0534	0535	0536	0537	0538	0539	0540	0541	0542	0543	0544	0545	0546	0547	0548	0549	0550	0551	0552	0553	0554	0555	0556	0557	0558	0559	0560	0561	0562	0563	0564	0565	0566	0567	0568	0569	0570	0571	0572	0573	0574	0575	0576	0577	0578	0579	0580	0581	0582	0583	0584	0585	0586	0587	0588	0589	0590	0591	0592	0593	0594	0595	0596	0597	0598	0599
0600	0601	0602	0603	0604	0605	0606	0607	0608	0609	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	0620	0621	0622	0623	0624	0625	0626	0627	0628	0629	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	0650	0651	0652	0653	0654	0655	0656	0657	0658	0659	0660	0661	0662	0663	0664	0665	0666	0667	0668	0669	0670	0671	0672	0673	0674	0675	0676	0677	0678	0679	0680	0681	0682	0683	0684	0685	0686	0687	0688	0689	0690	0691	0692	0693	0694	0695	0696	0697	0698	0699
0700	0701	0702	0703	0704	0705	0706	0707	0708	0709	0710	0711	0712	0713	0714	0715	0716	0717	0718	0719	0720	0721	0722	0723	0724	0725	0726	0727	0728	0729	0730	0731	0732	0733	0734	0735	0736	0737	0738	0739	0740	0741	0742	0743	0744	0745	0746	0747	0748	0749	0750	0751	0752	0753	0754	0755	0756	0757	0758	0759	0760	0761	0762	0763	0764	0765	0766	0767	0768	0769	0770	0771	0772	0773	0774	0775	0776	0777	0778	0779	0780	0781	0782	0783	0784	0785	0786	0787	0788	0789	0790	0791	0792	0793	0794	0795	0796	0797	0798	0799
0800	0801	0802	0803	0804	0805	0806	0807	0808	0809	0810	0811	0812	0813	0814	0815	0816	0817	0818	0819	0820	0821	0822	0823	0824	0825	0826	0827	0828	0829	0830	0831	0832	0833	0834	0835	0836	0837	0838	0839	0840	0841	0842	0843	0844	0845	0846	0847	0848	0849	0850	0851	0852	0853	0854	0855	0856	0857	0858	0859	0860	0861	0862	0863	0864	0865	0866	0867	0868	0869	0870	0871	0872	0873	0874	0875	0876	0877	0878	0879	0880	0881	0882	0883	0884	0885	0886	0887	0888	0889	0890	0891	0892	0893	0894	0895	0896	0897	0898	0899
0900	0901	0902	0903	0904	0905	0906	0907	0908	0909	0910	0911	0912	0913	0914	0915	0916	0917	0918	0919	0920	0921	0922	0923	0924	0925	0926	0927	0928	0929	0930	0931	0932	0933	0934	0935	0936	0937	0938	0939	0940	0941	0942	0943	0944	0945	0946	0947	0948	0949	0950	0951	0952	0953	0954	0955	0956	0957	0958	0959	0960	0961	0962	0963	0964	0965	0966	0967	0968	0969	0970	0971	0972	0973	0974	0975	0976	0977	0978	0979	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	0990	0991	0992	0993	0994	0995	0996	0997	0998	0999